

## **REMARKS**

Applicant respectfully requests reconsideration and allowance of all of the claims of the application. Claims 1, 3, 4, 6-12, 14, and 17-27 were pending at the time of the Office Action. Claims 20 and 22 are canceled without prejudice, claims 1, 6, 11, 12, 14, 17-19, 21, 23-24, 26 and 27 are amended, and new claims 28-31 are added herein. The amendments and the new claims are supported by the application as originally filed. For example, support for the amendments and the new claims may be found in the specification at least at pages 12-14 and Figure 5. Thus, no new matter is presented by the amendment. As a result, claims 1, 3, 4, 6-12, 14, 17-19, 21, and 23-31 are pending for consideration.

### **Cited Documents**

The following documents have been applied to reject one or more claims of the Application:

- **Dussud:** Dussud, U.S. Patent No. 6,502,111 has been applied to reject the claims of the Application.
- **[0007]:** Paragraph [0007] of the Background of the Specification

### **Claims 1, 3, 4, 6-12, 14, and 17-27 Are Non-Obvious Over Dussud**

Claims 1, 3, 4, 6-12, 14, and 17-27 stand rejected under 35 U.S.C. § 103(a) as allegedly being obvious over Dussud in view of [0007]. Applicant respectfully traverses the rejection. Nevertheless, solely for the purpose of expediting issuance, Applicant has amended claims 1, 6, 11, 12, 14, 17-19, 21, 23-24, 26 and 27 as shown above to highlight distinctions as discussed below. Moreover, cancellation renders moot the rejection of claims 20 and 22.

Independent Claim 1

In light of the amendments presented herein, Applicant submits that the rejection of independent claim 1 is moot. Specifically, the combination of Dussud and [0007] has not been shown to teach or suggest each element and feature in this claim as amended.

Amended claim 1 recites :

A computer-readable storage medium apparatus having computer-executable instructions encoded thereon encoded thereon to support ephemeral garbage collection by setting a write-watch mechanism to watch specified memory locations, the computer-readable storage medium apparatus being accessible by a computing device, the instructions when executed, configuring the computing device such that during execution of a program, when a statement of the program for execution is obtained, the computing device is configured to determine whether the statement includes a store operator, the computing device being further configured to perform operations comprising:

in an event the statement has a store operator:

storing a value specified in the statement in a memory location specified in the statement; and

determining whether the memory location specified is within an ephemeral generation;

in an event the memory location specified is within the ephemeral generation, obtaining a next statement of the program for execution;

in an event the memory location specified is not within the ephemeral generation, setting a card associated with the memory location specified and obtaining the next statement of the program for execution;

requesting via the write-watch mechanism a list of memory locations, the list:

identifying a plurality of the memory locations that have been accessed since a last ephemeral garbage collection process, each memory location corresponding to one of a plurality of cards associated with one or more objects allocated from within a memory heap, each of the plurality of cards associated with a card table, wherein the card table identifies one or more of the plurality of cards with objects that have been accessed; and

comprising a bitmap, wherein each bit within the bitmap corresponds to one of the plurality of cards, modification of the bitmap occurring when a corresponding bit is set at the time that the card is trimmed to disk;

creating, during the current ephemeral garbage collection process, a bundle table containing entries identifying a plurality of bundles, wherein each of the plurality of bundles identifies groupings of subsets of the plurality of cards;

marking, during the current ephemeral garbage collection process, two or more of the plurality of bundles identified in the bundle table using the list, wherein the marked bundles identify groupings of subsets of the plurality of marked cards having associated objects that have been accessed since the last ephemeral garbage collection process; and

performing garbage collection upon at least one accessed object.

Dussud is directed to concurrent garbage collection and describes that “modifications in the memory structure that occur during the concurrent marking act are logged ... by a write watch module. The application is then paused or stopped to perform a second marking act using information from the write watch module.” Dussud, Abstract.

[0007] discusses that “during garbage collection, the bundle bit map is checked to determine which bundles have objects that have been accessed. If the bundle bit map indicates that an object within the bundle has been accessed, each card in that bundle is checked to see if it has been accessed. If it has, then each of its objects is checked. While this technique improves the efficiency of the ephemeral garbage collection, the cost of executing the program is doubled. ... [T]he addition of [a] statement to perform bundling doubles the overhead for executing [a] store operator ... in comparison to only performing card-marking.” [0007], Application, p. 3.

Thus, Dussud describes concurrent garbage collection with two instances of marking, and [0007] discusses the problem of doubled overhead associated with two instances of marking. The purported combination of Dussud and [0007] would merely exemplify the problem

described in [0007]. In contrast, the claim removes a second instance of marking to an occurrence outside the loop so that the second instance of marking need not be concurrent.

Neither Dussud nor [0007], alone or in combination, have been shown to teach or suggest at least “support[ing] ephemeral garbage collection by setting a write-watch mechanism to watch specified memory locations, ... configuring the computing device such that during execution of a program, when a statement of the program for execution is obtained, the computing device is configured to determine whether the statement includes a store operator [and] in an event the statement has a store operator: storing a value specified in the statement in a memory location specified in the statement; and determining whether the memory location specified is within an ephemeral generation; in an event the memory location specified is within the ephemeral generation, obtaining a next statement of the program for execution; in an event the memory location specified is not within the ephemeral generation, setting a card associated with the memory location specified and obtaining the next statement of the program for execution; [and] requesting via the write-watch mechanism a list of memory locations the list.” as recited in amended claim 1.

Consequently, at least because not every element and feature recited in claim 1 is taught or suggested by the combination of Dussud and [0007], Applicant respectfully asserts that the pending rejection is overcome. Applicant respectfully requests that the rejection of this claim be withdrawn and the application passed to issuance.

*Dependent Claims 3, 4, 6-11, 28 and 29*

Claims 3, 4, 6-11, 28 and 29 ultimately depend from independent claim 1. As discussed above, independent claim 1 is patentable over the cited documents. Therefore, claims 3, 4, 6-11, 28 and 29 are also patentable over the cited documents of record for at least their

dependency from a patentable base claim. These claims may also be patentable for the additional features that each recites.

### Independent Claim 12

In light of the amendments presented herein, Applicant submits that the rejection of independent claim 12 is moot. Specifically, the combination of Dussud and [0007] has not been shown to teach or suggest each element and feature in this claim as amended.

Amended claim 12 recites:

A method for executing statements within a program to support ephemeral garbage collection by setting a write-watch mechanism to watch specified memory locations such that during execution of a program, when a statement of the program for execution including a store operator is obtained, a computing device is configured to perform the method comprising:

specifying a range of card table memory to watch during program execution by calling a write-watch mechanism that:

performs tracking of access to the card table memory; and

maintains a write-watch list that identifies cards written within the card table memory since a garbage collection process was last performed, each card being associated with and updated upon access to one or more objects allocated within a memory heap, the memory heap being divided into a plurality of cards with each card being grouped into one of a plurality of bundles, wherein one of the plurality of bundles corresponds to a subset of that plurality of cards that are tracked using a page of card table memory;

in an event the statement obtained has a store operator:

storing a value within the memory heap at a memory location specified by the statement obtained; and

determining whether the memory location specified is within an ephemeral generation;

in an event the memory location specified is within the ephemeral generation, obtaining a next statement of the program for execution;

in an event the memory location specified is not within the ephemeral generation, marking one of the plurality of cards within the card table memory corresponding to the memory location and obtaining a next statement of the program for execution.

As discussed above, Dussud describes concurrent garbage collection with two instances of marking, and [0007] discusses the problem of doubled overhead associated with two instances of marking. In contrast, the claim removes a second instance of marking to an occurrence outside the loop so that the second instance of marking need not be concurrent. At least for similar reasons as those discussed above, the purported combination of Dussud and [0007] would merely exemplify the problem described in [0007].

Neither Dussud nor [0007], alone or in combination, have been shown to teach or suggest at least “support[ing] ephemeral garbage collection by setting a write-watch mechanism to watch specified memory locations such that during execution of a program, when a statement of the program for execution including a store operator is obtained, a computing device is configured to ... in an event the statement obtained has a store operator: stor[e] a value within the memory heap at a memory location specified by the statement obtained; and determin[e] whether the memory location specified is within an ephemeral generation; in an event the memory location specified is within the ephemeral generation, obtain[] a next statement of the program for execution; in an event the memory location specified is not within the ephemeral generation, mark[] one of the plurality of cards within the card table memory corresponding to the memory location and obtaining a next statement of the program for execution,” as recited in amended claim 12.

Consequently, at least because not every element and feature recited in claim 12 is taught or suggested by the combination of Dussud and [0007], Applicant respectfully asserts that the pending rejection is overcome. Applicant respectfully requests that the rejection of this claim be withdrawn and the application passed to issuance.

Dependent Claims 14, 17, 18, 25, 26 and 30

Claims 14, 17, 18, 25, 26, and 30 ultimately depend from independent claim 12. As discussed above, independent claim 12 is asserted patentable over the cited documents. Therefore, claims 14, 17, 18, 25, 26, and 30 are also asserted patentable over the cited documents of record for at least their dependency from a patentable base claim. These claims may also be patentable for the additional features that each recites.

Independent Claim 19

In light of the amendments presented herein, Applicant submits that the rejection of independent claim 19 is moot. Specifically, the combination of Dussud and [0007] has not been shown to teach or suggest each element and feature of this claim as amended.

Amended claim 19 recites:

A memory management system configured to set a write-watch mechanism to watch specified memory locations during execution of a program, obtain a statement of the program for execution, determine whether the statement obtained includes a store operator, the system comprising:

a processor; and

a memory into which a plurality of instructions are loaded and into which a plurality of objects are dynamically allocated, the memory having a heap into which the objects are allocated, the heap being divided into a plurality of cards which are grouped into a plurality of bundles, each card being associated with one or more of the plurality of objects, wherein upon execution of the plurality of instructions by the processor, the system being configured to, based at least on whether the store operator is included in the statement for execution obtained, the system being further configured to perform an operation such that:

in an event the statement obtained does not have a store operator, executing the statement; and

in an event the statement obtained has a store operator:

storing a value specified in the statement obtained in a memory location specified in the statement obtained;

determining whether the memory location specified is within an ephemeral generation;

in an event the memory location specified is within the ephemeral generation, obtaining a next statement of the program for execution; and

in an event the memory location specified is not within the ephemeral generation, setting a card associated with the memory location specified and obtaining the next statement of the program for execution.

As discussed above, Dussud describes concurrent garbage collection with two instances of marking, and [0007] discusses the problem of doubled overhead associated with two instances of marking. In contrast, the claim removes a second instance of marking to an occurrence outside the loop so that the second instance of marking need not be concurrent. At least for similar reasons as those discussed above, the purported combination of Dussud and [0007] would merely exemplify the problem described in [0007].

Neither Dussud nor [0007], alone or in combination, have been shown to teach or suggest at least a system “configured to set a write-watch mechanism to watch specified memory locations during execution of a program, obtain a statement of the program for execution, ... determine whether the statement obtained includes a store operator [and] based at least on whether the store operator is included, the system being further configured to perform an operation such that: in an event the statement obtained does not have a store operator, executing the statement; and in an event the statement obtained has a store operator: storing a value specified in the statement obtained in a memory location specified in the statement obtained; determining whether the memory location specified is within an ephemeral generation; in an event the memory location specified is within the ephemeral generation, obtaining a next statement of the program for execution; and in an event the memory location specified is not within the ephemeral generation, setting a card associated with the memory location specified and obtaining the next statement of the program for execution” as recited in amended claim 19.



Consequently, at least because not every element and feature recited in claim 19 is taught or suggested by the combination of Dussud and [0007], Applicant respectfully asserts that the pending rejection is overcome. Applicant respectfully requests that the rejection of this claim be withdrawn and the application passed to issuance.

*Dependent Claims 21, 23, 24, and 31*

Claims 21, 23, 24, and 31 ultimately depend from independent claim 19. As discussed above, independent claim 19 is patentable over the cited documents. Therefore, claims 21, 23, 24, and 31 are also patentable over the cited documents of record for at least their dependency from a patentable base claim. These claims may also be patentable for the additional features that each recites.

## **Conclusion**

For at least the foregoing reasons, all pending claims are asserted in condition for allowance. Applicant respectfully requests reconsideration and prompt issuance of the application. If any issues remain that would prevent allowance of this application, **Applicant requests that the Examiner contact the undersigned representative before issuing a subsequent Action.**

Respectfully Submitted,

Lec & Hayes, PLLC  
Representatives for Applicant

By: /Bea Koempel-Thomas 58213/  
Beatrice L. Koempel-Thomas  
(bea@leehayes.com; 509-944-4759)  
Registration No. 58213

Dated: 05/24/2010

Andy M. Han  
(andyh@leehayes.com; 206-876-6006)  
Registration No. 60266